

What is claimed is:

- 1 1. A system for providing cooperative resource groups for high
2 availability applications, comprising:
3 a cluster framework including a plurality of nodes, comprising:
4 a plurality of cooperative resource groups, each comprising a
5 logical network address, at least one monitor and an application providing
6 services and externally accessed using the logical network address; and
7 a plurality of resources, each comprising a cluster service
8 supporting the services provided by each application;
9 for each cooperative resource group, a preferred node for execution and
10 one or more possible nodes as standby nodes for each other cooperative resource
11 group; and
12 each such cluster service restarting the services on a surviving node off a
13 critical path of the preferred node upon an unavailability of the preferred node,
14 while keeping the logical network address available on each possible node for the
15 cooperative resource group.
- 1 2. A system according to Claim 1, further comprising:
2 a run method starting each cooperative resource group in an ordered
3 fashion on a preferred node or on a possible node; and
4 a halt method stopping each cooperative resource group in an ordered
5 fashion on the node on which the halt method is running.
- 1 3. A system according to Claim 1, further comprising:
2 a watchdog process in one such cooperative resource group executing
3 upon a failure or shutdown of the cooperative resource group.
- 1 4. A system according to Claim 1, wherein the cluster service
2 operates in a normal mode with each cooperative resource group executing on the
3 preferred node.

1 13. A system according to Claim 1, wherein the cluster service
2 provides notification of a service halt by sending a service down event
3 notification from each preferred node.

1 14. A system according to Claim 1, wherein the cluster service
2 acquires an internet protocol address as the logical network address upon
3 executing the run method.

1 15. A method for providing cooperative resource groups for high
2 availability applications, comprising:
3 building a cluster framework including a plurality of nodes, comprising:
4 forming a plurality of cooperative resource groups, each
5 comprising a logical network address, at least one monitor and an application
6 providing services and externally accessed using the logical network address; and
7 structuring a plurality of resources, each comprising a cluster
8 service supporting the services provided by each application;
9 for each cooperative resource group, designating a preferred node for
10 execution and providing one or more possible nodes as standby nodes for each
11 other cooperative resource group; and
12 restarting the services on a surviving node off a critical path of the
13 preferred node upon an unavailability of the preferred node, while keeping the
14 logical network address available on each possible node for the cooperative
15 resource group.

1 16. A method according to Claim 15, further comprising:
2 executing a run method starting each cooperative resource group in an
3 ordered fashion on a preferred node or on a possible node; and
4 executing a halt method stopping each cooperative resource group in an
5 ordered fashion on the node on which the halt method is running.

1 17. A method according to Claim 15, further comprising:

2 spawning a watchdog process in one such cooperative resource group
3 upon a failure or shutdown of the cooperative resource group.

1 18. A method according to Claim 15, further comprising:
2 operating in a normal mode with each cooperative resource group
3 executing on the preferred node.

1 19. A method according to Claim 15, further comprising:
2 operating in an off-line mode with the logical network address available.

1 20. A method according to Claim 15, further comprising:
2 transferring the service off the critical path from the preferred node to one
3 such possible node responsive to a failover of the application in one such
4 cooperative resource group.

1 21. A method according to Claim 20, further comprising:
2 resuming the service off the critical path on another cooperative resource
3 group responsive to a failure or shutdown of the application; and
4 keeping the logical network address available on each possible node.

1 22. A method according to Claim 20, further comprising:
2 providing the logical network address of one such application upon a
3 failure or shutdown of the application.

1 23. A method according to Claim 15, further comprising:
2 transferring the service off the critical path to one such possible node
3 responsive to a switchover of the cooperative resource group.

1 24. A method according to Claim 15, further comprising:
2 creating a sequenced list of possible nodes for each cooperative resource
3 group.

1 25. A method according to Claim 24, further comprising:
2 disabling switching between the possible nodes for a last such possible
3 node for each cooperative resource group; and

4 issuing an alert.

1 26. A method according to Claim 15, further comprising:
2 providing notification of a service start by sending a service up event
3 notification from each preferred node.

1 27. A method according to Claim 15, further comprising:
2 providing notification of a service halt by sending a service down event
3 notification from each preferred node.

1 28. A method according to Claim 15, further comprising:
2 acquiring an internet protocol address as the logical network address upon
3 executing the run method.

1 29. A computer-readable storage medium holding code for performing
2 the method according to Claim 15.

1 30. A system for cooperatively clustering multiple instance
2 applications, comprising:
3 a node designated as a preferred node within a cluster framework
4 comprising a plurality of cooperative resource groups;
5 a cluster framework stack started on the preferred node, comprising:
6 an internet protocol address;
7 an application; and
8 application event monitors for the application; and
9 a run module sending notification to each other such cooperative resource
10 group within the cluster framework that the application is running and available
11 for service; and
12 a switching module enabling cooperative resource group switching from
13 the preferred node off a critical path for the application.

1 31. A system according to Claim 30, further comprising:

10086783 022802

2 at least one other node within the cluster framework designated as a
3 possible node, comprising acquiring a further internet protocol address for each
4 such possible node.

1 32. A system according to Claim 31, further comprising:
2 a cluster service executing within the cluster framework and restarting the
3 application off the critical path on the possible node responsive to one of a
4 failover and a switchover.

1 33. A system according to Claim 32, further comprising:
2 a halt method halting the application on the preferred node in parallel
3 responsive to the failover or the switchover, comprising releasing the further
4 internet protocol address.

1 34. A system according to Claim 33, further comprising:
2 a watchdog process on the preferred node upon the halting of the
3 application.

1 35. A system according to Claim 31, further comprising:
2 a halt module disabling cooperative resource group switching on a last
3 such possible node for each cooperative resource group.

1 36. A system according to Claim 30, further comprising:
2 a halt module halting the cluster framework stack, comprising stopping the
3 application event monitors, stopping the application and releasing the internet
4 protocol address; and
5 a run module sending notification to each other such cooperative resource
6 group within the cluster framework that the application is down and unavailable
7 for service.

1 37. A method for cooperatively clustering multiple instance
2 applications, comprising:
3 designating a node as a preferred node within a cluster framework
4 comprising a plurality of cooperative resource groups;

5 starting a cluster framework stack on the preferred node, comprising:
6 acquiring an internet protocol address;
7 starting an application; and
8 starting application event monitors for the application; and
9 sending notification to each other such cooperative resource group within
10 the cluster framework that the application is running and available for service; and
11 enabling cooperative resource group switching from the preferred node off
12 a critical path for the application.

1 38. A method according to Claim 37, further comprising:
2 designating at least one other node within the cluster framework as a
3 possible node, comprising acquiring a further internet protocol address for each
4 such possible node.

1 39. A system according to Claim 38, further comprising:
2 executing within the cluster framework and restarting the application off
3 the critical path on the possible node responsive to one of a failover and a
4 switchover.

1 40. A method according to Claim 39, further comprising:
2 halting the application on the preferred node in parallel responsive to the
3 failover or the switchover, comprising releasing the further internet protocol
4 address.

1 41. A method according to Claim 40, further comprising:
2 starting a watchdog process on the preferred node upon the halting of the
3 application.

1 42. A method according to Claim 38, further comprising:
2 disabling cooperative resource group switching on a last such possible
3 node for each cooperative resource group.

1 43. A method according to Claim 37, further comprising:
2 halting the cluster framework stack, comprising:

3 stopping the application event monitors;
4 stopping the application; and
5 releasing the internet protocol address; and
6 sending notification to each other such cooperative resource group within
7 the cluster framework that the application is down and unavailable for service.

1 44. A computer-readable storage medium holding code for performing
2 the method according to Claim 37.

20092230" 9849800T